

Pengenalan Karakter pada Plat Nomor Kendaraan Berbasis *Optical Character Recognition* Menggunakan Metode *Template Matching Correlation*

Yoga Aldian Putra¹, Catur Supriyanto²

^{1,2}Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Jl.Nakula I No.5-11, Semarang, Jawa Tengah, 50131, Indonesia

E-mail: 111201105952@mhs.dinus.ac.id ¹, catur.supriyanto@dsn.dinus.ac.id ²

Abstrak

Di era modern sekarang ini penggunaan kendaraan bermotor menunjukkan perkembangan yang sangat tinggi. Di samping itu kebutuhan akan otomatisasi pada otentikasi dirasa semakin perlu. Seiring semakin tingginya ilmu pengetahuan terutama di bidang komputer, maka semakin tinggi juga akan kebutuhan aplikasi komputer yang mampu mendukung dan meningkatkan kinerja manusia di berbagai bidang. Salah satunya yaitu aplikasi komputer yang dapat mendeteksi dan mengenali plat nomor kendaraan bermotor (mobil) untuk bidang keamanan. Dalam penelitian ini, akan dibuat aplikasi untuk mengenali plat nomor kendaraan berbasis OCR (*optical character recognition*). Citra yang berupa mobil akan dijadikan sebagai inputan program, citra tersebut kemudian diproses menggunakan deteksi kotak dengan tujuan memisahkan plat nomor yang ada pada citra dari *background*. Setelah didapatkan citra berupa plat nomor, langkah selanjutnya yaitu melakukan proses segmentasi karakter pada plat nomor kemudian hasilnya akan dikenali menggunakan metode *template matching correlation*. Dari uji coba yang dilakukan, didapat bahwa aplikasi ini dapat mengenali karakter yang ada di dalam plat nomor dengan tingkat keakuratan 80.29% dan dapat mengenali plat nomor dengan tepat dengan tingkat keakuratan 30%. Banyaknya jumlah data *template* sangat mempengaruhi dalam keakuratan pengenalan karakter.

Kata Kunci: otomatisasi, OCR (*optical character recognition*), *background*, *template matching correlation*, segmentasi karakter

Abstract

In this modern era the use of motor vehicle show a very high development. In addition, the need of automation in the authentication felt more necessary. As the higher of science especially in the computer field, the higher will be needs computer applications that can support and improve human performance in various fields. One of them is a computer application that can detect and identify the license plates of motor vehicles (cars) for the security field. In this study, the application will be made to recognize license plates on vehicle based OCR (optical character recognition). The image of a car will be used as an input of the program, then the image is processed using a detection box with the aim of separating the existing number plate in the image from the background. Once the image is obtained in the form of plates, the next step is to do the segmentation process of characters on the license plate then the result will be recognized using template matching correlation method. From the experiments performed, it was found that the application can recognize the characters in the license plate number with 80.29% accuracy rate and can recognize the number plate correctly with 30% accuracy. The number of template data affect the accuracy of character recognition.

Keywords: automation, OCR (*optical character recognition*), *background*, *template matching correlation*, character segmentation.

1. PENDAHULUAN

Di era modern sekarang ini penggunaan kendaraan bermotor menunjukkan perkembangan yang sangat tinggi. Di samping itu kebutuhan akan otomatisasi pada otentikasi dirasa semakin perlu [1]. Seiring semakin tingginya ilmu pengetahuan terutama di bidang komputer, maka semakin tinggi juga akan kebutuhan aplikasi komputer yang mampu mendukung dan meningkatkan kinerja manusia di berbagai bidang. Salah satunya yaitu aplikasi komputer yang dapat mendeteksi dan mengenali plat nomor kendaraan bermotor (mobil) untuk bidang keamanan.

Aplikasi pengenalan karakter pada plat nomor ini termasuk kedalam jenis *Optical Character Recognition* (OCR) yang merupakan jenis aplikasi untuk mengidentifikasi citra huruf dan angka kemudian diubah kedalam bentuk *text* [2]. Di dalam penelitian ini, akan dirancang sebuah program yang mampu mengenali plat nomor mobil berbasis OCR. Citra yang berupa mobil akan dijadikan sebagai inputan program, citra tersebut kemudian diproses dengan mendeteksi kontur objek yang berbentuk plat nomor pada citra untuk mendapatkan lokasi plat nomor dan kemudian plat akan dipisahkan dari *backgroundnya*. Pada citra plat nomor inilah proses ekstraksi objek karakter dilakukan kemudian dikenali dengan menggunakan metode *template matching correlation*.

Penelitian ini memiliki beberapa tujuan diantaranya yaitu untuk mengetahui bagaimana cara menerapkan metode OCR dalam pengenalan pola huruf dan angka pada plat nomor. Untuk mengetahui seberapa besar prosentase keberhasilan metode *template matching correlation* dalam mencocokkan suatu pola huruf dan angka.

2. METODE

Template matching correlation adalah metode pencocokan setiap piksel pada suatu matriks citra digital dengan citra yang menjadi acuan/*template* untuk menguji kedekatan sebuah data dengan yang lain. Proses pencocokan citra inputan dengan citra yang ada pada *template* akan menghasilkan nilai korelasi, tingkat korelasi yang tinggi menentukan suatu citra inputan tersebut dibaca/terkenali sebagai salah satu citra pada *template*. Nilai korelasi antara dua buah matriks yang terdiri dari citra inputan dan *template* diperoleh menggunakan persamaan [2]:

$$r = \frac{\sum_{k=1}^n (x_{ik} - \bar{x}_i) \cdot (x_{jk} - \bar{x}_j)}{\sqrt{\sum_{k=1}^n (x_{ik} - \bar{x}_i)^2 \cdot \sum_{k=1}^n (x_{jk} - \bar{x}_j)^2}} \quad (1)$$

$$\bar{x}_i = \frac{1}{n} \sum_{k=1}^n \bar{x}_{ik} \quad (2)$$

$$\bar{x}_j = \frac{1}{n} \sum_{k=1}^n \bar{x}_{jk} \quad (3)$$

Keterangan:

r = hasil korelasi dari dua buah matriks citra

\bar{x}_{ik} = nilai piksel ke- k dalam matriks i .

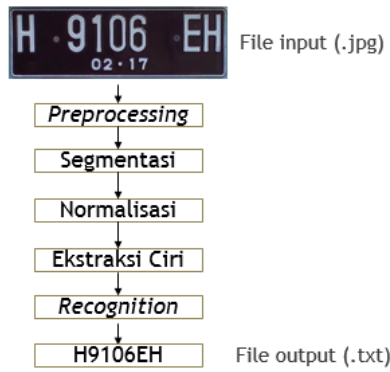
\bar{x}_{jk} = nilai piksel ke- k dalam matriks j .

\bar{x}_i = nilai rata-rata piksel pada citra i / matriks i .

\bar{x}_j = nilai rata-rata piksel pada citra j / matriks j .

n = total piksel yang ada pada suatu matriks.

Secara umum tahap-tahap yang dilakukan metode *template matching correlation* berbasis pada proses OCR yang terdiri dari *preprocessing*, segmentasi, normalisasi, ekstraksi ciri, *recognition* [3]. Gambaran umum dari proses OCR ditunjukkan dari gambar berikut:



Gambar 1. Proses OCR

2.1 Preprocessing

Preprocessing merupakan langkah awal dalam proses OCR. Beberapa proses yang ada dalam tahap *preprocessing* diantaranya yaitu, proses *grayscale*, deteksi tepi *canny*, binerisasi, dilasi, dan fill.

2.1.1 Grayscale

Citra *grayscale* merupakan citra beraras keabuan. Sedangkan *grayscale* yaitu proses merubah citra RGB kedalam citra keabuan dimana piksel yang awalnya mempunyai 3 nilai dirubah menjadi piksel yang mempunyai nilai tunggal menggunakan persamaan berikut [5]:

$$S = \frac{R+G+B}{3} \quad (4)$$

Nilai instensitas yang dihasilkan dari persamaan 4 memiliki nilai intensitas dari 0 sampai 255.

2.1.2 Binerisasi

Binerisasi yaitu suatu proses yang bertujuan untuk mengubah nilai intensitas yang ada pada citra menjadi 2 nilai yaitu 0 dan 1 yang disebut juga citra 2 bit. Dengan cara lain citra biner adalah citra yang hanya memiliki dua jenis warna yaitu hitam yang bernilai 0 dan putih yang bernilai 1. Proses binerisasi dapat dituliskan menggunakan persamaan berikut:

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) \geq T \\ 0 & \text{if } f(x, y) < T \end{cases} \quad (5)$$

Dengan $g(x, y)$ yaitu citra biner yang diperoleh dari citra *grayscale* $f(x, y)$, dan T merupakan nilai ambang/*threshold*.

2.2 Segmentasi

Segmentasi adalah pengelompokan piksel-piksel objek menjadi wilayah yang merepresentasikan suatu objek. Dalam penelitian ini segmentasi berfungsi untuk membagi / memotong setiap karakter yang terdapat pada plat nomor menjadi potongan-potongan citra karakter (huruf dan angka).

2.3 Normalisasi

Normalisasi yaitu proses merubah citra inputan yang bertujuan untuk menyamakannya dengan citra pada data template. Proses normalisasi yang digunakan dalam penelitian ini yaitu penyamaan ukuran citra. Pada saat proses pencocokkan pola menggunakan algoritma *template matching correlation*, ukuran dari citra inputan harus disesuaikan dengan ukuran citra yang ada di dalam data template.

2.3.1 Penskalaan

Merupakan suatu operasi geometri yang bertujuan untuk memperbesar atau memperkecil ukuran citra inputan sesuai dengan variabel penskalaan citranya. Hasil penskalaan diperoleh dari perkalian antara ukuran citra input dengan variabel penskalaan. Proses penskalaan dapat dilakukan dengan persamaan[4]:

$$P_o = S_p \times P_i \quad (6)$$

$$L_o = S_i \times L_i \quad (7)$$

Dimana (P_i, L_i) adalah satuan ukuran citra input, (P_o, L_o) adalah ukuran citra output, dan (S_p, S_i) adalah variabel penskalaan yang diinginkan. Jika variabel penskalaan lebih besar dari 1

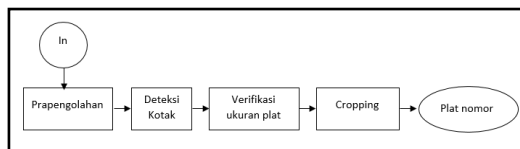
maka hasil penskalaannya akan memperbesar ukuran citra, sebaliknya apabila variabel penskalaannya lebih kecil dari 1 maka hasilnya akan memperkecil ukuran citra.

3. HASIL DAN PEMBAHASAN

Secara umum sistem terbagi menjadi dua bagian utama yaitu pendeteksian lokasi plat nomor dan pengenalan plat nomor.

3.1 Pendeteksian Lokasi Plat Nomor

Tahap pendeteksian plat nomor berfungsi untuk menemukan lokasi plat nomor dan kemudian memisahkan plat nomor dari background, sehingga didapatkan citra yang berupa plat nomor. Proses pendeteksian plat nomor ditunjukkan pada gambar



Gambar 2. Proses deteksi plat nomor

Proses *preprocessing* dilakukan untuk menyesuaikan apa yang dibutuhkan untuk proses selanjutnya. Dalam proses *preprocessing* terdapat beberapa tahapan yaitu *grayscale*, deteksi tepi canny, binerisasi, penghapusan piksel, dilasi dan *fill*. Yang pertama, sistem akan merubah objek RGB kedalam bentuk *grayscale* dengan menggunakan persamaan 4. Dengan menerapkan persamaan tersebut ke dalam setiap piksel pada citra, maka semua piksel akan berubah menjadi satu jenis warna (keabuan) dengan nilai intensitas yang berbeda-beda.

Kemudian citra akan dilakukan deteksi tepi canny sehingga citra berubah menjadi tepian dari objek. Berikut merupakan langkah yang terdapat dalam pendeteksian tepi:

- a. Langkah pertama adalah membuang piksel (*noise*) pada citra dengan

menerapkan Gaussian *blur*, yang membuat citra tampak sedikit buram. Proses tersebut bertujuan untuk memperoleh tepian dari citra yang sebenarnya. Jika proses tersebut tidak diterapkan hasilnya garis-garis halus akan terdeteksi juga sebagai tepian dari objek.

- b. Langkah kedua yaitu melakukan deteksi tepi menggunakan deteksi tepi sobel. Berikut ini merupakan salah satu contoh dari deteksi tepi sobel:

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

G_x G_y

Hasil perhitungan tersebut kemudian disatukan untuk memperoleh hasil gabungan tepi vertikal dan horizontal dengan menggunakan persamaan:

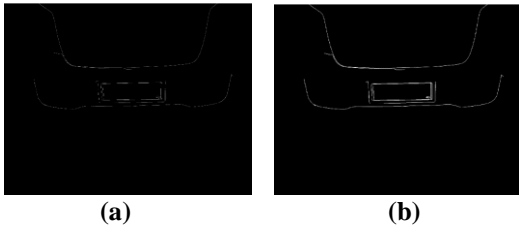
$$|G| = |G_x| + |G_y| \quad (8)$$

- c. Langkah ketiga yaitu menentukan arah dari tepian yang terdeteksi, kemudian membaginya kedalam empat warna sehingga masing-masing garis yang mempunyai arah yang beda akan mempunyai warna berbeda.
- d. Langkah keempat yaitu menerapkan *non maximum suppression* pada garis yang muncul sehingga diperoleh garis tepian dari objek yang lebih ramping. Langkah terakhir dari deteksi tepi canny yaitu binerisasi dengan menerapkan 2 buah nilai *threshold*. Gambar berikut merupakan citra sebelum dilakukan deteksi tepi dan citra sesudah pemrosesan.



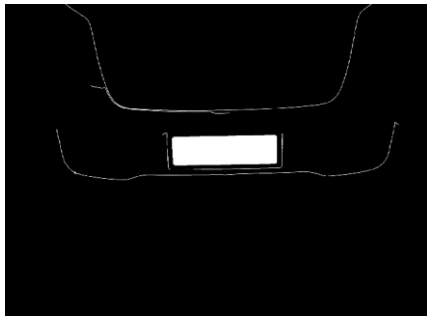
(a) (b)
Gambar 3. (a) Citra masukan (b)
Hasil deteksi tepi canny

Setelah citra berubah menjadi tepian dari objek, kemudian dilakukan penghapusan piksel untuk mengurangi noise, dengan menggunakan aturan 8-connectivity, piksel yang saling berhubungan dengan jumlah kurang dari 900 akan dihapus. Untuk mengurangi resiko garis yang membentuk objek kotak terputus digunakan operasi dilasi/penumbuhan dengan bentuk (*strel*) *diamond*. Setiap piksel yang terdapat pada citra dilakukan penumbuhan sehingga menjadi seperti gambar berikut:



(a) (b)
Gambar 4. (a) Hasil penghapusan piksel (b)
Hasil dilasi

Garis yang saling berhubungan kemudian akan diberi *fill*, sehingga menjadi seperti pada gambar 5.



Gambar 5. Hasil fill

Dalam proses pengambilan plat nomor, diambil dari objek yang berhasil diberi

fill, kemudian diseleksi lagi dengan batasan ($\text{lebar} > \text{tinggi}$ dan $\text{diagonal} \geq 0.7$). Sehingga didapat objek berupa plat nomor.



Gambar 6. Citra plat nomor

3.2 Pengenalan karakter

Setelah didapatkan citra berupa plat nomor, langkah selanjutnya plat nomor akan dirubah kedalam bentuk biner dan untuk mengurangi noise, piksel yang saling berhubungan kurang dari 200 piksel akan dihilangkan dengan menggunakan aturan 8-connectivity.



Gambar 7. Citra plat nomor biner

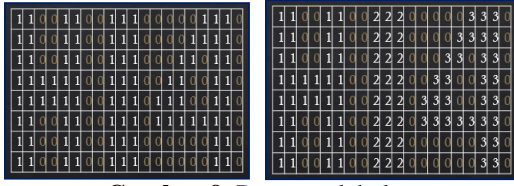
Karena objek yang akan dikenali hanya nomor dari plat, maka plat nomor akan dipotong untuk memisahkan dari bingkai kotak dan tanggal berlaku plat nomor.



Gambar 8. Cropping plat nomor

Kemudian plat nomor akan disegmentasi menjadi potongan karakter yang ada didalam plat nomor. Langkah segmentasi karakter adalah sebagai berikut:

- Menggunakan fungsi pelabelan dengan aturan 8-connectivity. Pelabelan digunakan untuk mengelompokkan setiap piksel yang saling terhubung menurut aturan 8-connectivity. Setiap komponen terhubung yang saling tidak terhubung (*disjoin*) pada suatu citra akan diberi (label) berbeda[6].



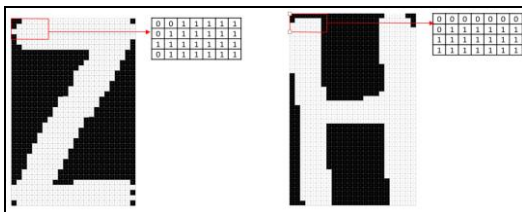
Gambar 9. Proses pelabelan

- b. Menggunakan fungsi untuk menampung koordinat pada masing-masing piksel yang telah diberi label berdasarkan baris dan kolom.
- c. Setelah semua koordinat piksel ditampung, *crop* piksel dengan label yang berbeda-beda. Koordinat piksel yang dipakai untuk *crop* terdiri dari 4 piksel yaitu (piksel teratas, piksel terbawah, piksel terkanan dan piksel terkiri) dari masing-masing label.



Gambar 10. Proses deteksi tepi canny

Semua citra hasil segmentasi kemudian diresize dengan ukuran 24x42 piksel untuk menyamakan ukurannya dengan data template. Langkah terakhir citra akan dikenali dengan metode *template matching correlation* dan hasilnya akan disimpan kedalam .txt. *Template matching correlation* merupakan metode pencocokkan setiap piksel pada suatu matriks citra digital dengan citra yang menjadi acuan (*template*) untuk menguji kesamaan sebuah data dengan yang lain pada saat pengenalan yang dihitung dari kesamaan antar dua buah matriks citra (nilai korelasi). Citra hasil segmentasi akan di bandingkan dengan setiap data *template* (A-Z, 9-0) untuk dicari nilai korelasi tertingginya dengan menggunakan persamaan 1:



Gambar 11. Korelasi Antara inputan H dengan *template Z*

$$r = \frac{\sum_{k=1}^n (x_{ik} - \bar{x}_i) \cdot (x_{jk} - \bar{x}_j)}{\sqrt{\sum_{k=1}^n (x_{ik} - \bar{x}_i)^2 \cdot \sum_{k=1}^n (x_{jk} - \bar{x}_j)^2}} =$$

$$\frac{(0-0.4871) \cdot (0-0.5347)}{\sqrt{(0-0.4871)^2 \cdot (0-0.5347)^2}} + \frac{(0-0.4871) \cdot (0-0.5347)}{\sqrt{(0-0.4871)^2 \cdot (0-0.5347)^2}}$$

$$+ \frac{(1-0.4871) \cdot (0-0.5347)}{\sqrt{(1-0.4871)^2 \cdot (0-0.5347)^2}} + n$$

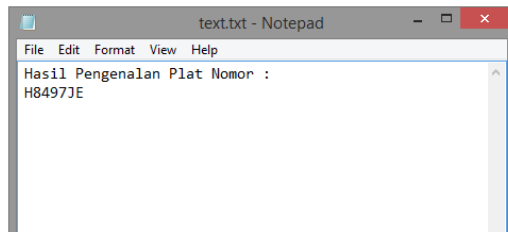
Persamaan diatas merupakan contoh dari perhitungan *template matching correlation*, karena ukuran dari gambar yang digunakan adalah 24x42 piksel, maka persamaan tersebut dijumlahkan sebanyak 1008 kali.

Berikut adalah hasil korelasi gambar 10 dengan semua data *template*.

Tabel 1: Hasil korelasi

Karakter	Korelasi Tertinggi	Kolom ke-	Terbukti
H	0. 7146	8	✓
8	0. 6375	34	✓
4	0. 4264	30	✓
9	0. 6084	35	✓
7	0. 7318	33	✓
J	0. 7659	10	✓
E	0. 5738	5	✓

Tabel 1 menunjukkan hasil korelasi karakter yang ada pada gambar 10 dengan masing-masing *template* (A-Z, 9-0). Kolom kedua menunjukkan nilai korelasi tertinggi dan kolom ketiga menunjukkan letak dari nilai korelasi tertinggi. Setelah hasil segmentasi dibandingkan dengan semua data *template* dan diketahui nilai korelasinya, kemudian dicari pada kolom berapakah nilai korelasi menunjukkan hasil tertinggi, setelah diketahui lokasi baris kemudian program akan menampilkan isi dari baris tersebut ke dalam .txt.



Gambar 12. Hasil pengenalan karakter (*.txt)

3.3 Hasil Pengujian

Implementasi dari penggunaan algoritma *template matching correlation* digunakan untuk proses pengenalan karakter dari hasil segmentasi plat nomor sebelumnya. Hasil segmentasi plat nomor yang telah dilakukan ekstraksi ciri akan dihitung nilai korelasinya dengan masing-masing data *template* untuk mengenali karakter.

Tabel 2: Hasil pengujian

No.	Plat Nomor	Teridentifikasi
1	H7633TY	H7633TY
2	H9106EH	H91 <u>D</u> 6EH
3	H9193KG	H9193K <u>C</u>
4	H7635WY	H <u>Y</u> 635 <u>V</u> Y
5	H9348PR	H9348 <u>B</u> <u>B</u>
6	H7545LG	H7545L <u>C</u>
7	H8857YH	H <u>3</u> 857YH
8	AA9065QK	AA9 <u>Q</u> 65 <u>D</u> K
9	H8854CY	H8854CY
10	H33DN	H3 <u>J</u> DN

Dari pengujian dataset sebanyak 40 buah, dapat diperoleh nilai akurasi dan *error rate* dalam pengenalan karakter dengan perhitungan:

$$akurasi = \frac{220}{274} \times 100\% = 80.29\%$$

$$error\ rate = \frac{54}{274} \times 100\% = 19.71\%$$

Untuk mengetahui nilai akurasi dan *error rate* dalam pengenalan plat nomor

dengan tepat digunakan perhitungan:

$$akurasi = \frac{12}{40} \times 100\% = 30\%$$

$$error\ rate = \frac{28}{40} \times 100\% = 70\%$$

4. KESIMPULAN DAN SARAN

Dari penelitian yang dilakukan dapat ditarik suatu kesimpulan dan saran.

4.1 Kesimpulan

Kesimpulan yang didapat dari penelitian ini adalah

1. Implementasi dari deteksi kotak, dan *template matching correlation* dapat mengenali karakter pada citra plat nomor mobil.
2. Akurasi pengenalan karakter sebesar 80.29% dan *error rate* sebesar 19.71%
3. Akurasi pengenalan plat nomor dengan sempurna sebesar 30% dan *error rate* sebesar 70%.
4. Jika plat nomor telah mengalami modifikasi pada bentuk karakternya, maka semakin kecil keakuratan dalam proses pengenalan.
5. Banyaknya data *template* yang digunakan dalam proses pengenalan sangat mempengaruhi akurasi ketepatan dalam mengenali karakter.

4.2 Saran

Dari kesimpulan yang didapat, penulis memberi saran hal-hal yang perlu diperhatikan untuk penelitian kedepannya:

1. Akurasi pengenalan plat nomor dengan tepat sebesar sebesar 30% dirasa masi rendah. Diharapkan pada penelitian selanjutnya dapat digunakan metode lain yang dapat meningkatkan akurasi pengenalan untuk mengurangi efek negatif dari pengenalan plat nomor.
2. Pengenalan plat nomor dapat diaplikasikan tidak hanya untuk plat nomor kendaraan pribadi yang

mempunyai warna dasar hitam dan warna karakter putih tetapi juga kendaraan umum dengan warna dasar kuning dan warna karakter hitam serta plat nomor kendaraan dinas dengan warna dasar merah dan warna karakter putih.

3. Diharapkan dalam penelitian selanjutnya pengenalan citra plat nomor dapat juga diaplikasikan untuk sepeda motor atau roda 2 dan juga dapat mendeteksi objek lebih dari 1.

DAFTAR PUSTAKA

- [1] R. R. Arief, "*Analisis Penggunaan Scale Invariant Feature Transform Sebagai Metode Ekstraksi Fitur Pada Pengenalan Jenis Kendaraan*" Skripsi Teknik Elektro Universitas Indonesia, Depok, 2010.
- [2] Rezki Trianto, Ni Nengah Dewi Merdekawati, Rizkie Purnama Shakti Nugraha, dkk, "*Klasifikasi Huruf Katakana Dengan Metode Template Matching Correlation*" 2010.
- [3] Suryo Hartanto, Aris Sugiharto, Sukmawati Nur Endah, "*Optical Character Recognition Menggunakan Algoritma Template Matching Correlation*" Journal of Informatics and Technology, vol. 1, pp. 11-20, 2012.
- [4] T Sutoyo, Edy Mulyanto, Vincent Suharono, Oky Dwi Nurhayati, and Wijanarto, *Teori Pengolahan Citra Digital*, Semarang: Andi, 2009.
- [5] W. R. Prasetya, "*Pengenalan Plat Nomor Kendaraan Secara Otomatis Untuk Pelanggaran Lalu Lintas*".
- [6] D. Putra, *Pengolahan Citra Digital*, Yogyakarta: Andi, 2010.